

Software Engineering 1

Abgabedokument

Teilaufgabe 1

(Anforderungsanalyse und Planungsphase)

Persönliche Daten, bitte vollständig ausfüllen:

Nachname, Vorname:	Badera, Sofiia
Matrikelnummer:	11715248
E-Mail-Adresse:	a11715248@unet.univie.ac.at
Datum:	27.10.2023

Aufgabe 1: Anforderungsanalyse

Analysieren der Spielidee und des Netzwerkprotokolls um 8 Anforderungen (bestehend zumindest aus 3 funktionalen, 3 nichtfunktionalen und einer Designbedingung) nach den folgenden Kriterien zu dokumentieren. Achten Sie darauf den im Skriptum und der Vorlesung behandelten Qualitätsaspekten Genüge zu tun.

Typ der Anforderung: funktional

Anforderung 1

- **Beschreibung:** Registrierung des Clients – Nach Start des Spiels muss Client KI sich am Server registrieren.
- **Bezugsquelle:** Spielidee, **Nach Start des Clients registrieren sich die KIs für das Spiel am Server**

Anforderung 2

- **Beschreibung:** Eine Burg pro Kartenhälfte – Nach Registrierung des Clients muss Client KI die Kartenhälfte mit genau einer Burg generieren.
- **Bezugsquelle:** Netzwerkprotokoll, **Im Anschluss an die Registrierung eines Spielers muss der Client eine Kartenhälfte generieren und an den Server übertragen. Hierbei wird jedes Feld individuell definiert, basierend auf dem Typ des Feldterrains sowie einer möglicherweise vorhandenen Burg. Wichtig hierbei ist, dass auf mindestens einem Feld eine Burg positioniert sein muss, gleichzeitig, aber auch nicht mehr wie ein Feld gleichzeitig eine Burg beinhalten darf.**

Anforderung 3

- **Beschreibung:** Client sendet Bewegung an Server – Nach Kartenhälfte austausch muss Client die Bewegung an Server senden um Schatz zu finden.
- **Bezugsquelle:** Spielidee, **Um den Schatz zu finden, bewegen beide KIs ihre Spielfigur über die Karte und decken dabei mit ihrer Spielfigur schrittweise Kartenfelder auf.**

Typ der Anforderung: nicht funktional

Anforderung 4

- **Beschreibung:** Optimale Anzahl der Spiele auf Server – Parallel müssen maximal 99 Spiele auf Server ausgeführt werden um die Ressourcen des Servers zu schonen.
- **Bezugsquelle:** Netzwerkprotokoll, **Um die Ressourcen des Servers zu schonen gilt die Einschränkung, dass maximal 99 Spiele parallel ausgeführt werden dürfen.**

Anforderung 5

- **Beschreibung:** Optimale Bedenkzeit der KI – Während ihre Runde soll KI maximal 5 Sekunden Bedenkzeit haben um die Spiele für die Zuschauer spannend zu gestalten.
- **Bezugsquelle:** Spielidee, **Um die Spiele für die Zuschauer spannend zu gestalten, wurde festgelegt, dass ein Spiel insgesamt nicht länger als 320 Spielaktionen (und damit 320 Runden) dauern darf. Für jede dieser rundenbasierten Spielaktion hat die KI maximal 5 Sekunden Bedenkzeit.**

Anforderung 6

- **Beschreibung:** Optimale Zeitspanne zwischen Spielstatusabfragen – Zwischen Spielstatusabfragen eines Clients muss es mindestens 0,4 Sekunden sein um Server nicht zu überlasten.
- **Bezugsquelle:** Netzwerkprotokoll, **Um zu verhindern, dass der Server überlastet wird muss zwischen zwei vom gleichen Client durchgeführten Abfragen zum Spielstatus mindestens eine Zeitspanne von 0,4 Sekunden vergehen.**

Typ der Anforderung: Designbedingung

Anforderung 7

- **Beschreibung:** Format der ausgetauschten Nachrichten – Die ausgetauschten Daten bzw. Nachrichten zwischen Client und Server müssen im XML Format definiert werden.
- **Bezugsquelle:** Netzwerkprotokoll, **Die ausgetauschten Daten bzw. Nachrichten werden im XML Format definiert bzw. erwartet.**

Anforderung 8

- **Beschreibung:** Technologie des Nachrichtenaustauschs – Nachrichtenaustausch zwischen Client und Server muss via HTTP Protokoll ausgeführt werden.
- **Bezugsquelle:** Netzwerkprotokoll, **Die technologische Basis des Nachrichtenaustauschs stellt eine Restschnittstelle dar, daher es wird das HTTP Protokoll verwendet sowie die zugehörigen Operationen GET und POST.**

Aufgabe 2: Anforderungsdokumentation

Dokumentation einer zum relevanten Bereich passenden Anforderung nach dem vorgegebenen Schema. Ziehen Sie eine Anforderung heran, für die alle Bestandteile der Vorlage mit relevantem Inhalt befüllt werden können. Wir empfehlen hierzu eine **funktionale** Anforderung auszuwählen.

Dokumentation Anforderung

- **Name:** Client sendet eine Bewegung an Server.
- **Beschreibung und Priorität:** Nach Kartenhälftenaustausch muss Client die Bewegung an Server senden um Schatz zu finden.
Priorität: Hoch.
- **Relevante Anforderungen:**
 - Generierung der Bewegung – Nach Kartenaustausch muss Client KI die Bewegungen erstellen.
 - Validierung der Bewegung – Als Server die Bewegung bekommt, muss er die Relevanz der Bewegung auswerten.
 - Aktualisierung der Spieldaten – Server muss die passende Spieldaten für die Clients aktualisieren.
 - Terminierung der Client Aktionen - Im Anschluss muss Client keine Aktionen (Kartenhälften, Bewegungen) mehr an den Server senden.
 - Änderung der Bewegungsrichtung – Jederzeit kann Client KI eine neue Bewegungsrichtung wählen.
- **Relevante Business Rules:**
 - Wasser und Wiesen können mit einer Bewegung betreten werden.
 - Berge können mit zweien Bewegungen betreten werden.
 - Berge können mit zweien Bewegungen verlässt werden.
 - Wann Client ein Wasser Feld betrete, muss er verlieren.
 - Wann Client die Karte flüchtet, muss er verlieren.
- **Impuls/Ergebnis - Typisches Szenario:**
Vorbedingungen:
 - Ein Menschliche Spieler erstellt ein neues Spiel an dem Server.
 - Server registriert der Client für das Spiel.
 - Client KI generiert eine Kartenhälfte.
 - Client KI stellt eine Burg auf der Kartenhälfte.
 - Beide Clients senden die Kartenhälften an den Server.
 - Server validiert die Kartenhälften.
 - Server erstellt die Spielkarte.
- **Hauptsächlicher Ablauf:**
 - Client fragt die Spielstatus ab.
 - Server antwortet mit „MustAct“ Status.
 - Client generiert die Bewegung.
 - Client sendet seine Bewegung an Server.
 - Server sendet die Bestätigung.

Nachbedingungen:

- Server aktualisiert die Status und Position von dem Spieler.
- Client sendet nächste Bewegung.
- Das Spiel wird enden, wenn ein Client gewinnt oder verliert.

• **Impuls/Ergebnis - Alternativszenario:**

Vorbedingungen:

- Ein Menschliche Spieler erstellt ein neues Spiel an dem Server.
- Server registriert der Client für das Spiel.
- Client KI generiert eine Kartenhälfte.
- Client KI stellt eine Burg auf der Kartenhälfte.
- Beide Clients senden die Kartenhälften an den Server.
- Server validiert die Kartenhälften.
- Server erstellt die Spielkarte.

Hauptsächlicher Ablauf:

- Client fragt die Spielstatus ab.
- Server antwortet mit „MustWait“ Status.
- Client sendet Spielstatus Abfragen jede 0,4 Sekunde.
- Server antwortet mit „MustAct“ Status.
- Client Response dauert länger als 5 Sekunden.

Nachbedingungen:

- Client fragt die Spielstatus ab.
- Server antwortet mit „Lost“ Status.
- Der Menschliche Spieler wird über Verlust informiert.

Impuls/Ergebnis - Fehlerfall:

Vorbedingungen:

- Ein Menschliche Spieler erstellt ein neues Spiel an dem Server.
- Server registriert der Client für das Spiel.
- Client KI generiert eine Kartenhälfte.
- Client KI stellt eine Burg auf der Kartenhälfte.
- Beide Clients senden die Kartenhälften an den Server.
- Server validiert die Kartenhälften.
- Server erstellt die Spielkarte.

Hauptsächlicher Ablauf:

- Client fragt die Spielstatus ab.
- Server antwortet mit „MustAct“ Status.
- Client generiert die Bewegung.
- Client sendet seine Bewegung an Server.
- Server sendet die Error Antwort mit Exception, weil die Bewegung in Richtung Wasserfeld war.

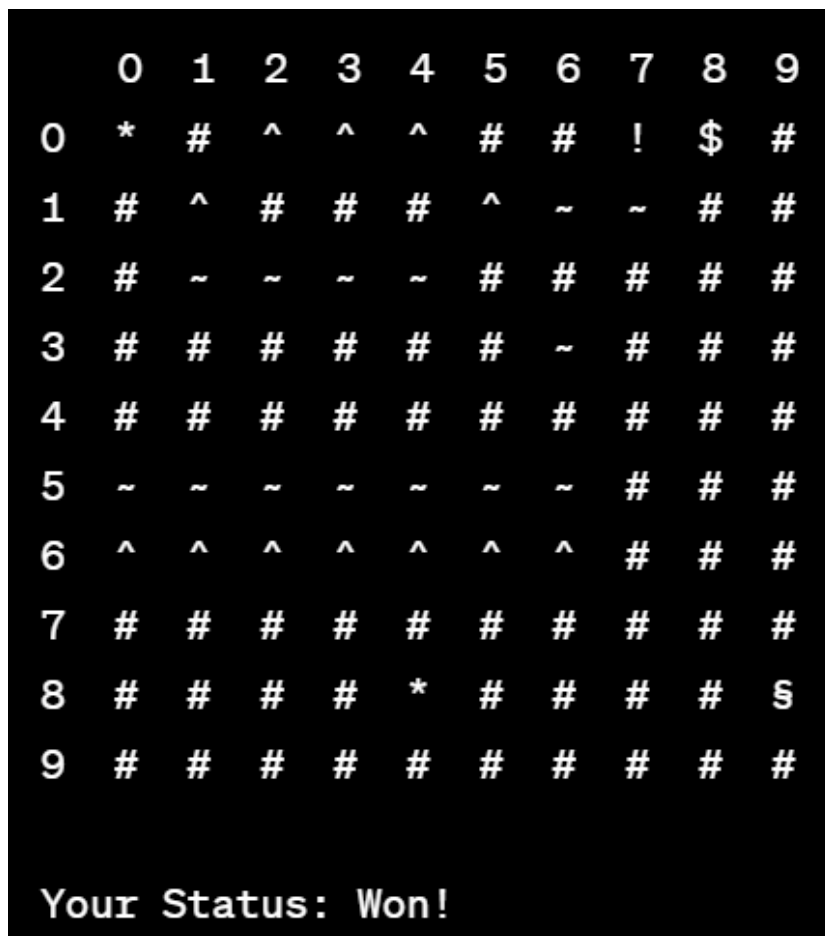
Nachbedingungen:

- Client fragt die Spielstatus ab.
- Server antwortet mit „Lost“ Status.
- Der Menschliche Spieler wird über Verlust informiert.

• **Benutzergeschichten:**

- Als Menschlicher Anwender möchte ich mein Client KI die Bewegung effektiv zu generieren und an Server zu senden, um das Spiel zu gewinnen.
- Als Server möchte ich Client die Bewegung richtig und rechtzeitig an mich zu senden, um das Spiel weiterlaufen zu können.

• **Benutzerschnittstelle:**

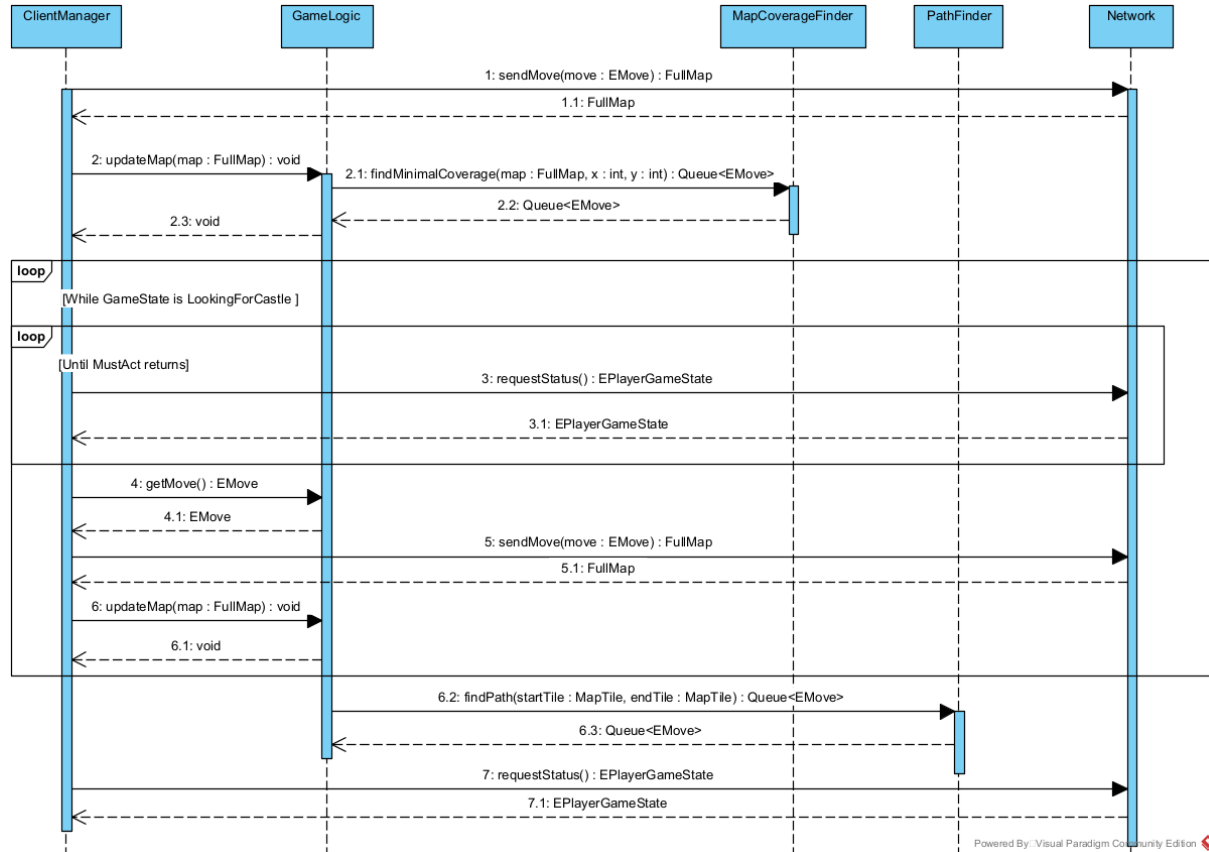


• **Externe Schnittstellen:**

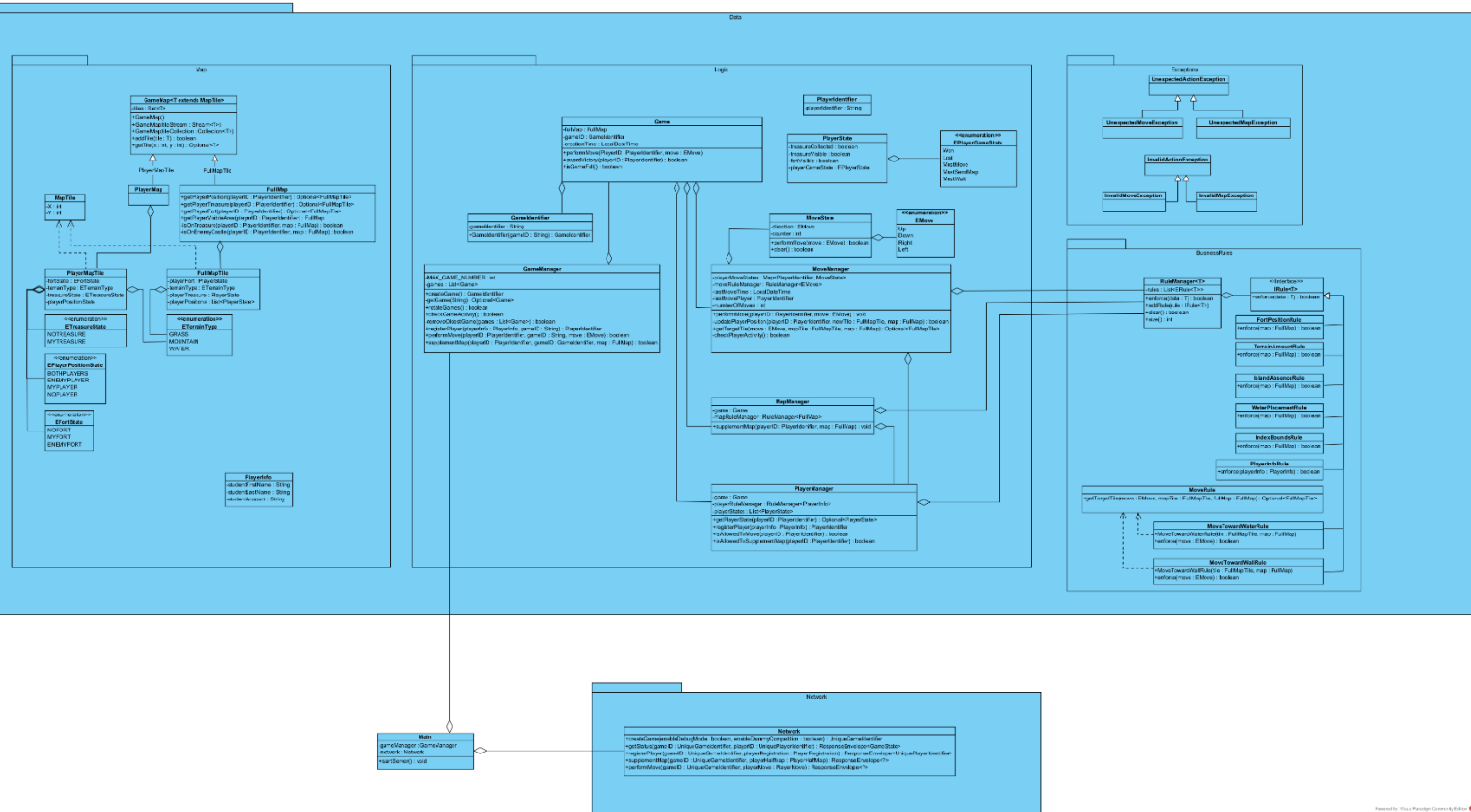
- o Technologie des Nachrichtenaustauschs wird mit HTTP Protokoll ausgeführt.
- o Nachrichten zwischen Client und Server werden im XML Format erwartet.

Client UML Sequenzdiagramm

Ablauf bis zum endgültigen Spielende ausgehend von folgendem angenommenen Spielstand. Spielstand: Ihr Client hat seinen Schatz und befindet sich auf einem Berg neben der gegnerischen Burg auf die Aktion des gegnerischen Clients wartend.



Server UML Klassendiagramm



Server UML Sequenzdiagramm

Das vollständige empfangen, verarbeiten und beantworten einer über den /halfmaps Endpoint einlangenden Kartenhälfte. Letztere wurde von einem Client versandt der zuvor schon eine Kartehälfte und Bewegungen zum Spiel beigetragen hat.

